**Naval Research Laboratory**

**Washington, DC 20375-5320**

**NRL/MR/7230--04-8759**

# NRL Atmospheric Correction Algorithms for Oceans: Land, Cloud, and Cirrus *Mask* User's Guide

Marcos J. Montes
Bo-Cai Gao

*Coastal and Ocean Remote Sensing Branch*
*Remote Sensing Division*

Curtiss O. Davis

*Remote Sensing Division*

March 22, 2004

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



**1. REPORT DATE** *(DD-MM-YYYY)*
March 22, 2004

**2. REPORT TYPE**
Memorandum

**3. DATES COVERED** *(From - To)*
January 2002-December 2003

**4. TITLE AND SUBTITLE**
NRL Atmospheric Correction Algorithms for Oceans: Land, Cloud, and Cirrus *Mask* User's Guide

**5a. CONTRACT NUMBER**

**5b. GRANT NUMBER**

**5c. PROGRAM ELEMENT NUMBER**
61153N

**5d. PROJECT NUMBER**

**5e. TASK NUMBER**

**5f. WORK UNIT NUMBER**

**6. AUTHOR(S)**
Marcos J. Montes, Bo-Cai Gao, and Curtiss O. Davis

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Naval Research Laboratory, Code 7230
4555 Overlook Avenue, SW
Washington, DC 20375-5320

**8. PERFORMING ORGANIZATION REPORT NUMBER**
NRL/MR/7230--04-8759

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5660

**10. SPONSOR / MONITOR'S ACRONYM(S)**

**11. SPONSOR / MONITOR'S REPORT NUMBER(S)**

**12. DISTRIBUTION / AVAILABILITY STATEMENT**
Approved for public release; distribution is unlimited.

**13. SUPPLEMENTARY NOTES**

**14. ABSTRACT**
Many ocean color processing algorithms often need to have an additional module to mask out the non-oceanic objects, such as clouds or land. We have constructed a program that produces land, cirrus, low-altitude cloud, and bad-pixel masks. This document describes the usage of the executable application derived from these algorithms.

**15. SUBJECT TERMS**
Atmospheric correction; Hyperspectral remote sensing

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| **a. REPORT** | **b. ABSTRACT** | **c. THIS PAGE** | | | Marcos J. Montes |
| Unclassified | Unclassified | Unclassified | UL | 16 | **19b. TELEPHONE NUMBER** *(include area code)* (202) 767-7308 |

**Standard Form 298 (Rev. 8-98)**
**Prescribed by ANSI Std. Z39.18**

# Contents

# 1 Background

Many ocean color processing algorithms often need to have an additional module to mask out the non-oceanic objects, such as clouds or land. We have constructed a program known as *Mask* which produces land, cirrus, low altitude cloud, and bad pixel masks. While the land mask appears to be fairly robust, the cirrus and low-altitude cloud masks clearly need some more work, and there are plans to update them.

# 2 The Masking Algorithms

In this document, we'll frequently use the observed "apparent" reflectance $\rho^*_{\text{obs}}$ defined as

$$\rho^*_{\text{obs}} \equiv \frac{\pi L_{\text{t}}}{\mu_0 E_0} \ , \tag{1}$$

where $L_{\text{t}}$ is the observed (i.e., at the sensor) radiance; $E_0$ is the extraterrestrial solar irradiance; and $\mu_0 = \cos\theta_0$, where $\theta_0$ is the solar zenith angle. In general, reflectance is a spectral quantity and a subscript is assumed in all equations. All of the equations below will, however, explicitly refer to a specific wavelength.

## 2.1 Bad Pixel Mask

A bad pixel is defined as any pixel that has

$$\text{all}_{i=1}^{N_B}(N_i \leq M)$$

where $i$ indexes the band number and bands are numbered from 1 to $N_B$, the maximum number of bands; $N_i$ is the scaled digital number used to represent the radiance for band $i$. The value $M$ may be set using the keyword `bad_pixel_value = M`, where $M$ is the value chosen by the user. When this keyword is not present, the default value is $M = 0$. This mask is always produced, and is always the last band in the output mask file.

If the bad pixels are on the edge of the image, it may be easier to edit the data and remove the offending pixels. The values for the keywords `x start` and `samples` should be modified appropriately when this is done in order to keep a record of the actual sample number (which is useful for geo-correction purposes, for example). These keywords are recognized by *Mask*, and will allow the user to keep a record of the true sample number included in the image. Similarly, *Mask* will also recognize the keywords `y start` and `lines`.

## 2.2 Land Mask

There are two possible land masks: a generalized "Normalized Difference" mask (described in §2.2.1), and a mask utilizing a strict reflectance limit at a particular wavelength (described in §2.2.2). In some cases, either of these may work; in others, neither of the methods may work well. In both cases, the user may exercise many options in constructing the land mask.

We note that a pixel that is only partially filled by land (i.e., bridges over water, or boats in water that are of sub-pixel dimensions) tends *not* to be flagged as being land, thus the user will need to pay attention to such instances. It is also the case that the same pixel, when atmospherically corrected, will definitely have a spectrum that does not appear to be consistent with water, and so should be fairly easy to deal with.

The "Normalized Difference Index" land mask described in §2.2.1 is the default land mask used.

### 2.2.1 Normalized Difference Index Land Mask

The default land mask is the the "Normalized Difference Vegetative Index", i.e., *NDVI*, as our discriminator between land and water[1]. *NDVI* is defined as

$$NDVI \equiv \frac{\rho_{\text{obs}}(0.86\mu\text{m}) - \rho_{\text{obs}}(0.66\mu\text{m})}{\rho_{\text{obs}}(0.86\mu\text{m}) + \rho_{\text{obs}}(0.66\mu\text{m})} \ . \tag{2}$$

Manuscript approved March 5, 2004.

[1] This NDVI is not consistent with the AVHRR time series unless the bands and instrumental response of AVHRR are used, and will be biased with respect to it.

After some experimentation, we have set the default discriminator as

$$NDVI > 0.05 \text{ implies "Land"} . \tag{3}$$

However we have left some generality by allowing the user to set the value with the `ndvi_land_mask_threshold` keyword, as will be explained below. Additionally, the user may select the wavelengths to use with the `ndvi_wavelengths` (the default is {`0.66, 0.86`}), and `ndvi_nbands` is a two element integer array which describes the number of bands to use in determining the reflectance at the given wavelengths (the elements need not be identical, but they may be). A value of `0` implies that a linear interpolation of the two nearest bands will be performed in order to determine the reflectance at the desired wavelength; a positive integer `N` implies that the $N$ nearest bands (to the desired wavelength) will be averaged in order to determine the reflectance to be used in the previous equations. The default value is {`0,0`}, which implies linear interpolation to determine the value at both of the input wavelengths.

It is usually the case that over water we find that $\rho_{\rm obs}(0.86\mu{\rm m}) < \rho_{\rm obs}(0.66\mu{\rm m})$, thus the *NDVI* is negative. While this may not be true in some shallower waters, or in waters with high sediment loads, our cutoff at $NDVI > 0.05$ appears to be sufficient in most cases.

#### 2.2.2 Reflectance Limit Land Mask

Another way to calculate a land mask may be to just have a strict apparent reflectance limit at a particular wavelength, i.e,

$$\rho_{\rm obs}(\lambda) > \rho_{\rm limit} \text{ implies "Land"} . \tag{4}$$

This version is implemented when `refl_land_mask_threshold`$\geq 0$, and the wavelength used is the *second* wavelength in the `ndvi_wavelengths` real array.

#### 2.2.3 General Comments about Land Mask Algorithms

The ability to choose the wavelengths for either of the above land masks is important when the data is noisy or the spectra do not extend all the way to 0.865 $\mu$m. It is possible, however, that the reflectance limit land mask will get confused by the presence of clouds and cirrus, and thus falsely identify them as land; it is less likely that this will happen with NDVI land masks. This confusion is not a problem as long as the masks are used solely to flag pixels that will not be processed.

In all instances, the wavelengths chosen should be in atmospheric "window" regions, i.e, regions with little atmospheric gas absorption. For the first wavelength, $\lambda \sim 660$ nm is a good choice. For the second wavelength, regions such as $\lambda \sim 750$ nm, 865 nm, and other window regions are useful. It is also advantageous that the second wavelength occur in a region that has a significant increase in absorption by water, as this will help increase the contrast between water and land. No internal checks are done to ensure that appropriate spectral regions are chosen, so the user can construct fairly arbitrary masks in this manner.

No matter which mask is requested, the *NDVI* values (or, more correctly, the values of the particular normalized-difference index that is requested using the values associated with the keyword `ndvi_wavelengths`) as well as the reflectance values at the two indicated wavelengths are all output to the *NDVI* output file. Examining this output will allow a user to better choose values for a mask, and experiment with different normalized-difference indices utilizing a range of combinations of wavelengths and numbers of bands.

### 2.3 Cirrus Mask

The cirrus mask is based on the work Gao et al. (see, for example, Gao and Goetz [1993], Gao and Kaufman [1995], Gao et al. [1998]), and uses the 1.375 $\mu$m channel to detect the presence of cirrus clouds. Currently the algorithm uses a simple threshold, so that

$$\rho_{\rm obs}(1.375\mu{\rm m}) > \rho_{1.375 \text{ threshold}} \tag{5}$$

indicates the presence of cirrus clouds. The quantity $\rho_{1.375 \text{ threshold}}$ is an input that is set with the keyword `cirrus_mask_threshold`. When the keyword is not entered by the user, the default value of $\rho_{1.375 \text{ threshold}} = 0.0025$ is used.

The calculation of $\rho_{\rm obs}(1.375\mu{\rm m})$ is affected by the value of `cirrus_mask_nbands`. A value of `0` will get the default behavior of linear interpolation; a positive value will average that number of the nearest values; negative numbers will bring the program to a halt.

Experience has shown us that the limit of 0.0025 is perhaps too low, at least over land. A more sophisticated approach than the one described by Equation 5 is necessary, with at least two thresholds, one over land, and one over water. Some recent research using cloud and cirrus masks for hyperspectral imagers may be found in Mandl et al. [2003a,b], as well as experience with cirrus masks used by the MODIS instruments (which involves a ratio of $\rho^*_{\rm obs}(1.375\mu{\rm m})/\rho^*_{\rm obs}(1.24\mu{\rm m})$) may be incorporated into *Mask* in the future.

### 2.4 Low-Altitude Cloud Mask

The low-altitude cloud mask is not expected to be all that robust, and is only guaranteed to give even possibly satisfactory results over dark areas, such as water. It is not expected to work at all over land. In fact, no determination of low altitude clouds is currently made over land. The value

$$\rho_{\rm obs}(0.94\mu{\rm m}) > \rho_{0.94\ {\rm threshold}} \tag{6}$$

indicates the presence of low-altitude clouds over water. This threshold value is selected by the user with the keyword `cloud_mask_ocean_threshold`. When the keyword is not entered by the user, the default value of $\rho_{0.94\ {\rm threshold}} = 0.1$ is used.

The method for calculating $\rho_{\rm obs}(0.94\mu{\rm m})$ is controlled by the value of `cloud_mask_nbands`. A value of `0` will get the default behavior of linear interpolation; a positive value will average that number of the nearest values; negative numbers will bring the program to a halt.

Again, a more robust low-cloud mask is much more difficult and time consuming. More work will be done to produce such a mask at a later time. Some recent research using cloud and cirrus masks for hyperspectral imagers may be found in Mandl et al. [2003a,b], and may be incorporated into *Mask* in the future.

## 3 Running *Mask*

### 3.1 Source Code

*Mask* has been created from a single[2] source code that can be compiled on at least three platforms. Currently, it has been compiled on a PC platform with the *Microsoft® Fortran PowerStation™ 4* compiler, on an SGI® workstation with the *MIPS® Pro 7.3 Fortran 90* compiler, and on a Sun® workstation with the *Sun® Forte™ Fortran 95 version 6 update 2* compiler. All attempts have been made to make the source code strictly ANSI/ISO Fortran 90 [ISO/IEC 1539:1991] compliant. Since none of the deleted features are used, it is also compliant with standards of Fortran 95 [ISO/IEC 1539-1:1997] and Fortran 2003 [ISO/IEC 1539-1:2004]. Language extensions beyond Fortran 77 [ISO/IEC 1539:1978] are used, so the source code will not compile with a standard Fortran 77 compiler. The executable should run correctly when compiled with an ANSI standard Fortran 90 compiler that: 1) assumes that unformatted direct access files are "flat" binary data files with no record length information or record marks in the file[3]; 2) supports the reading and writing of 1, 2, and 4 byte integers[4]; 3) supports a common numerical model[5] in order to allow for distribution of the accompanying binary data tables.

Mask will run very fast since the tests are performed for a whole line of data at a time using the Fortran 90 `where` constructs.

### 3.2 Invocation

To invoke or execute the *Mask*, it is best to feed it the input file via a "standard input redirect," which is allowed under both the command prompt window on a PC, and under most shells available on the

[2]This is *not quite* true. A single file called `name_and_version.f90` differs on each platform, usually only with the text identifying the hardware although sometimes the `version_date` may differ.

[3]This is not specified in the Fortran 90 standard, but is supported by many vendors.

[4]Only one integer representation is required in a standard conforming Fortran 90 processor, but more are allowed; many compilers support three or four types of integers.

[5]The method of representing floating point numbers or integers using binary digits.

preponderance of Unix based operating systems. Thus, if I symbolize the prompt as `prompt>` , the following is valid on both PCs and under `csh` & `tcsh`:
`prompt> path-to-mask < mask_input_file`
Use the full path to both the *Mask* executable and the *Mask* input file in order to avoid accessing out-of-date versions of both.

The easiest way to run *Mask* on the currently supported platforms is to use the *ENVI* interface provided by W. Snyder[6]. The graphical interface gathers the required information from the user, and prepares the image header file and the *Mask* input file.

## 3.3 *Mask* Inputs

The *Mask* input file is an ASCII text file made up of many keywords. The structure is

`keyword = value`

The values come in six basic types: integer, real, and string; integer array, real array, and string array. Array types are enclosed in curly braces, i.e., { }.

Except for `mask_input_image_name`, the keywords may appear in either the input file or the image's *ENVI* header file. However, it is considered good form to include keywords that concern information required to read the data or concerning the actual image, to appear in the image's header file. Keywords specific to *Mask* should then appear in the input file.

Any lines that appear in the input file must have no more than 132 characters in the line; any values that must wrap around (such as long arrays) should have line breaks *after* a comma, after a left curly brace, or before a right curly brace.

Certain other keywords have been defined, and may occur in the header file. The lists here do *not* include all these keywords. Only the keywords *required* by *Mask* are listed below. *Mask* will ignore other text in the file.

If a keyword is repeated in the input file and in the header file, the value in the input file is the value that will be used.

In the follwing pages, items in the `teletype` font style are meant to represent what you would actually expect to see or enter in a file. Any other text, in particular, text in parentheses, is a useful comment indicating restrictions on value choices, implied units, suggested reasonable values, etc.

### 3.3.1 Input Image Header File

With that preferred breakdown, the following keywords should appear in the image's header file (the values are examples only). An example input image header file is listed in Appendix A.1.2.

- `samples = 614`
- `lines = 600`
- `bands = 224`
- `header offset = 0`
- `data type = 2` (2 = signed 2 byte integer, which *Mask* requires)
- `interleave = bip` (`bip`, `bil`, or `bsq`.)
- `byte order = 1` (0 which is Least Significant Byte first, for PC/DEC; else 1 which is Most Significant Byte First)
- `wavelength = { 0.4121, ..., 2.5090}` (microns)
- `fwhm = { 0.009691, ..., 0.010030}` (microns)

[6]Please use the contact information on the front cover of this document in order to find out more about the *ENVI* interface.

The following items should also appear in the image's header file, since it is the logical location for them. These keywords are non-standard for any image display application. In some instances, the keywords may disappear if the header file is edited from within the image display application; in other instances, the image display application will merely ignore any keywords it does not understand. Because of this, an argument can made for them to be put in the input file. Current practice is to include them in the image's header file, and remind the user to be cautious when editing the header file from within any image processing application.

- `image_scale_factor= { 100., ..., 50.}` (Either 1 element if all `bands` elements have the same scale factor, or `bands` elements if they don't. These are the numbers we divide the integers in the file by in order to obtain the physical, measured radiance in units of W $m^{-2}$ $sr^{-1}$ $\mu m^{-1}$.
- `image_center_date = {yyyy, mm, dd}` (GMT date, assuming standard civil Gregorian calendar, $1 \leq$ `mm` $\leq 12$, and $1 \leq$ `dd` $\leq 31$.)
- `image_center_time = {hh, mm, ss.ss}` (GMT time, $0 \leq$ `hh` $< 24$; $0 \leq$ `mm` $< 60$; $0 \leq$ `ss.sss` $< 60$)
- `image_center_long = {ddd, mm, ss.sss}` (non-negative, degrees, minutes, seconds; $0. \leq$ `ddd` $\leq 180.$)
- `image_center_long_hem = W` (Either `W` or `E` of the Prime Meridian)
- `image_center_lat = {dd, mm, ss.sss}` (non-negative, degrees, minutes, seconds; $0. \leq$ `dd` $\leq 90.$)
- `image_center_lat_hem = N` (Either `N` or `S` of the Equator)
- `image_center_zenith_ang = {dd, mm, ss.sss}` (non-negative)
- `image_center_azimuth_ang = {dd, mm, ss.sss}` (non-negative, relative to north, clockwise)

### 3.3.2 *Mask* Input File

The following items are specific to *Mask*, and thus should appear in the *Mask* input file. An example input file listed in Appendix A.1.1.

- `mask_input_image_name = /u1/mmontes/coral.bip`
- `mask_data_directory = /u1/mmontes/data/` (The directory that contains the high resolution exoatmospheric solar irradiance spectrum in units of W $m^{-2}$ $sr^{-1}$ $\mu m^{-1}$. The data is in a binary file named `sun_binary` that consists of reals and is expected to be in the native byte order.)
- `mask_output_root_name = /u1/mmontes/testing` (The name, including directory, to which is appended various suffices in order to form the various output image and header file names.)
- `mask_which_masks = {land, cirrus, low cloud}` (This string array must include one or more of the set `land`, `cirrus`, or `low cloud`.)
- `ndvi_land_mask_threshold = 0.05` ($NDVI >$ `ndvi_land_mask_threshold` implies land. This is an optional keyword, and you'll get the default value if this keyword is not included.)
- `land_mask_threshold` (This is an obsolete version of `ndvi_land_mask_threshold`. This is kept solely for backward compatibility, and may disappear in future versions. If both appear in a file and have different values, only `ndvi_land_mask_threshold` is used.)
- `ndvi_wavelengths = { 0.66, 0.86 }` (An optional keyword consisting of a 2-element real array. These wavelengths are used to determine the normalized-difference index. If this keyword is not present, the default values of 0.66 and 0.86 micron are used.)
- `ndvi_nbands = { 0, 0 }` (An optional keyword consisting of a 2-element integer array. The value 0 implies that the reflectance for the *NDVI* calculation is formed by linear interpolation of neighboring values; otherwise, a positive integer indicates an average over that many of the nearest bands to the wavelength indicated in `ndvi_wavelengths`. The default is `{0,0}`).

- refl_land_mask_threshold = 0.045 (An optional keyword. If positive, it implies that the reflectance test (§2.2.2) is used to determine the land mask, and ndvi_land_mask_threshold is ignored; if non-positive, then the normalized difference test (§2.2.1) is used instead.)

- cirrus_mask_threshold = 0.0025 ($\rho_{\rm obs}(1.375\mu{\rm m}) >$ cirrus_mask_threshold implies the presence of cirrus clouds. This is an optional keyword, and you'll get the default value if this keyword is not included.)

- cirrus_mask_nbands = 0 (An optional integer keyword; 0 implies a linear interpolation is used to calculate $\rho_{\rm obs}(1.375\mu{\rm m})$, a positive number indicates that number of bands is averaged in order to calculate the aforementioned quantity. The default is 0.)

- cloud_mask_ocean_threshold = 0.1
($\rho_{\rm obs}(0.94\mu{\rm m}) >$ cloud_mask_ocean_threshold implies the presence of cirrus clouds. This is an optional keyword, and you'll get the default value if this keyword is not included.)

- cloud_mask_nbands = 0 (An optional integer keyword; 0 implies a linear interpolation is used to calculate $\rho_{\rm obs}(0.94\mu{\rm m})$, a positive number indicates that number of bands is averaged in order to calculate the aforementioned quantity. The default is 0.)

### 3.4 Output Files

A mask image file, along with a header is always output. When a land mask is requested, an *NDVI* file and its header file are also output. Both header files will contain a "history" section to indicate the version and settings used to generate them.

The mask file names are:

- mask_output_root_name_mask.hdr &
- mask_output_root_name_mask.img .

The mask image contains up to four planes of BSQ interleaved byte data. The mask values are either 0 (not masked) or 100 (masked). An example output header file is listed in Appendix A.2.1.

The *NDVI* file names are:

- mask_output_root_name_ndvi.hdr &
- mask_output_root_name_ndvi.img .

This file contains three planes: *NDVI* values, and the contituent reflectances of the calculations. The data are four byte floating point numbers in the native byte order. In practice the file may not contain *NDVI* values, but some other normalized difference index, depending on the values of the input parameters. An example *NDVI* header file is listed in Appendix A.2.2.

## 4 Frequent Answers: Useful Notes & Items to Consider Before Using *Mask*

This section represents wisdom culled from the experience previous uses of *Mask*, as well as behavior that is not documented elsewhere in this *User's Guide*.

1. The ordering of the data affects the speed of execution. All else being equal, BIP data is processed the fastest, and BSQ data is processed the slowest. Data ordering is a compromise of all uses of the data, however. *Mask* probably executes fast enough that the sort order (the interleave) of the data does not matter for the particular data set that you use. As the data set becomes larger, the absolute time difference of *Mask* runs on data of different interleaves will increase.

2. The byte order of the data affects the speed of execution. All else being equal, data in the byte order that is the same as the native byte order of the computer you are using will execute faster than if *Mask* needs to byte-swap the data as it is read. *Mask* probably executes fast enough that the time difference does not matter for your particular data set.

3. ASCII text files (such as header files and *Mask* input files) need to be carefully transferred between Unix and non-Unix platforms, as the end of line character differs. Ensure that FTP transfers of text files is performed using ASCII mode. Take the time to open the text files and verify that there are no strange characters at the end of the each line.

4. Ensure that the last line of the header and tafkaa input files has an end-of-line appropiate to your operating system. Or, better yet, just add a blank line at the end of an ASCII text file that *Mask* will be reading.

5. *Mask* is case sensitive when reading keywords and values. Ensure you are using the correct case as shown in the sections above. There is a difference between the number one (`1`) and the lower-case letter "L" (`l`). There is also a difference between the number zero (`0`), and the upper- and lower-case "O" (`O` and `o`, respectively). If *Tafkaa* does not find a keyword that you know is in the header or input file, check each keyword for correct spelling.

6. With operating systems, hardware, or compilers that use signed 32-bit addressing there is no way to address files that are larger than $2^{31} - 1 = 2147483647$ bytes, $\approx$ 2.1GB (where kB = $10^3$B, not 1024B). This issue is most noticeable for some PCs. Many recent computers do not have this problem since they use signed 64-bit integers, which implies a maximum file size of $2^{63} - 1$ bytes, $\approx$ 9.2EB. There are other addressing schemes in use that allow access to files larger than $\approx$ 2.1GB, but still smaller than $\approx$ 9.2EB. Users that apply *Mask* to files larger than the operating system can handle will get unpredictable results that depend on the hardware, operating system, and probably the version of the compiler that was used to make the executable. Past symptoms have been that *Mask* just seems to stop processing (sometimes with exiting, sometimes without) at a location (calculated from line number, sample, and bands) $\approx$ 2.1 GB into the input file.

## Acknowledgments

The development of *Mask* and of this *User's Guide* was supported by the Office of Naval Research.
*Mask* was generated using modified versions of subroutines from B.-C. Gao's *ATREM.*
Many thanks to Bill Snyder for helpful suggestions for new features, and for finding and reporting bugs.
This document was typeset using LaTeX 2ε.

## Legal



## Bibliography

B.-C. Gao and A. F. H Goetz. Cirrus Cloud Detection from Airborne Imaging Spectrometer Data Using the 1.38 $\mu$m Water Vapor Band. *Geo. Res. Lett.*, 20(4):301—304, February 1993.

B.-C. Gao and Y. J. Kaufman. Selection of the 1.375 $\mu$m MODIS Channel for Remote Sensing of Cirrus Clouds and Stratospheric Aerosols from Space. *J. Atmos. Sci.*, 52(23):4231–4237, December 1995.

B.-C. Gao, Y. J. Kaufman, W. Han, and W. J. Wiscombe. Correction of thin path radiances in the 0.4–1.0 $\mu$m spectral region using the sensitive 1.375 $\mu$m cirrus detecting channel. *J. Geophys. Res.*, 103(D24): 32169–32176, December 1998.

ISO/IEC 1539:1978. Fortran 77.

ISO/IEC 1539:1991. Fortran 90.

ISO/IEC 1539-1:1997. Information technology - Programming languages - Fortran - Part 1: Base Language. Fortran 95.

ISO/IEC 1539-1:2004. Information technology - Programming languages - Fortran - Part 1: Base Language. Fortran 2003.

D. Mandl, M. Griffin, H. Burke, and J. Miller. Validation of On-board Cloud Cover Assessment Using EO-1. ESTO Science and Technology Conference, at `http://esto.nasa.gov:8080/conferences/estc2003/papers/A3P2(Mandl).pdf`, June 2003a.

D. Mandl, J. Miller, and M. Griffin. EO-1 Onboard Cloud Cover Detection Validation Presentation. ESTO Science and Technology Conference, at `http://esto.nasa.gov:8080/conferences/estc2003/Presentations/A3/A3P2(Ma%ndl).pdf`, June 2003b.

# A Example Header Files

I've decided to include full examples of input files, input image header files, and output header files. About the only difference between the files as shown here and the files as they exist on the computer is the necessary existence of page breaks when the files are displayed as I have chosen to display them here. There are no page breaks in the input files on the computer.

## A.1 Input Files

### A.1.1 *Mask* Input File

```
mask_input_image_name = /usr/people/mmontes/atrem_cois/coral.bip
mask_data_directory = ../
mask_output_root_name = ./coral_test
mask_which_masks = { land, cirrus, low cloud}
ndvi_land_mask_threshold = 0.05
ndvi_wavelengths = { 0.66, 0.86 }
ndvi_nbands = { 0, 0 }
ndvi_scale_factor = 10000.
cirrus_mask_threshold = 0.0045
cirrus_mask_nbands = 0
cloud_mask_ocean_threshold = 0.1
cloud_mask_nbands = 0
```

### A.1.2 *ENVI* Style Input Radiance Image Header File

Based on the input file in Appendix A.1.1, the name of the header file below should be `/usr/people/mmontes/atrem_cois/coral.hdr`. However, if that name is not found, *Mask* will also search for the name `/usr/people/mmontes/atrem_cois/coral.bip.hdr`. As can be seen from the history portion of the file below, it was generated on PC using W. Snyder's header population widget from his *ENVI* interface routines. Additionally, I have limited the `wavelength` and `fwhm` entries to only $\leq 90$ characters wide so they can fit the page.

```
ENVI
description = {
File Resize Result, x resize factor: 1.0000, y resize factor: 1.0000. [Thu
Sep 11 17:21:43 1997]}
samples = 614
lines = 600
bands = 224
header offset = 0
interleave = bip
data type = 2
byte order = 1
file type = ENVI Standard
sensor type = AVIRIS
y start = 298
image_unscaled_units = W/m2/micron/ster
image_center_date = {1996, 3, 23}
image_center_time = {19, 44,   29.000}
image_center_long = {81, 47,   54.000}
image_center_long_hem = W
image_center_lat = {24, 36,   44.000}
image_center_lat_hem = N
image_center_zenith_ang = {0, 0,    0.000}
image_center_azimuth_ang = {0, 0,    0.000}
sensor_altitude = 19.768000
image_scale_factor = {
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
50, 50, 50, 50, 50, 50,
100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100,
100, 100, 100, 100, 100, 100, 100, 100, 100, 100}
wavelength = { 0.369850, 0.379690, 0.389530, 0.399370, 0.409210, 0.419060, 0.428910,
0.438760, 0.448610, 0.458460, 0.468310, 0.478170, 0.488020, 0.497880, 0.507740, 0.517600,
0.527470, 0.537330, 0.547200, 0.557070, 0.566940, 0.576810, 0.586680, 0.596560, 0.606430,
0.616310, 0.626190, 0.636070, 0.645960, 0.655840, 0.665730, 0.675620, 0.664140, 0.673720,
0.683300, 0.692880, 0.702460, 0.712040, 0.721630, 0.731210, 0.740800, 0.750380, 0.759970,
0.769560, 0.779140, 0.788730, 0.798320, 0.807910, 0.817500, 0.827090, 0.836680, 0.846280,
0.855870, 0.865470, 0.875060, 0.884660, 0.894250, 0.903850, 0.913450, 0.923050, 0.932640,
0.942240, 0.951850, 0.961450, 0.971050, 0.980650, 0.990250, 0.999860, 1.009460, 1.019070,
1.028680, 1.038280, 1.047890, 1.057500, 1.067110, 1.076720, 1.086330, 1.095940, 1.105550,
1.115160, 1.124780, 1.134390, 1.144010, 1.153620, 1.163240, 1.172860, 1.182470, 1.192090,
1.201710, 1.211330, 1.220950, 1.230570, 1.240190, 1.249820, 1.259440, 1.269060, 1.254320,
1.264280, 1.274240, 1.284200, 1.294150, 1.304110, 1.314070, 1.324030, 1.333990, 1.343950,
1.353910, 1.363870, 1.373820, 1.383780, 1.393740, 1.403700, 1.413660, 1.423610, 1.433570,
1.443530, 1.453490, 1.463440, 1.473400, 1.483360, 1.493310, 1.503270, 1.513230, 1.523180,
1.533140, 1.543100, 1.553050, 1.563010, 1.572970, 1.582920, 1.592880, 1.602830, 1.612790,
1.622740, 1.632700, 1.642650, 1.652610, 1.662560, 1.672520, 1.682470, 1.692430, 1.702380,
1.712340, 1.722290, 1.732250, 1.742200, 1.752150, 1.762110, 1.772060, 1.782020, 1.791970,
```

```
1.801920, 1.811880, 1.821830, 1.831780, 1.841730, 1.851690, 1.861640, 1.871590, 1.881540,
1.878300, 1.888340, 1.898390, 1.908430, 1.918460, 1.928500, 1.938530, 1.948560, 1.958590,
1.968620, 1.978640, 1.988670, 1.998690, 2.008700, 2.018720, 2.028730, 2.038740, 2.048750,
2.058760, 2.068760, 2.078770, 2.088770, 2.098770, 2.108760, 2.118760, 2.128750, 2.138740,
2.148720, 2.158710, 2.168690, 2.178670, 2.188650, 2.198630, 2.208600, 2.218570, 2.228540,
2.238510, 2.248480, 2.258440, 2.268400, 2.278360, 2.288320, 2.298270, 2.308220, 2.318170,
2.328120, 2.338070, 2.348010, 2.357950, 2.367890, 2.377830, 2.387760, 2.397690, 2.407620,
2.417550, 2.427480, 2.437400, 2.447320, 2.457240, 2.467160, 2.477080, 2.486990, 2.496900,
2.506810}
fwhm = { 0.009610, 0.009580, 0.009550, 0.009530, 0.009500, 0.009480, 0.009460, 0.009440,
0.009420, 0.009400, 0.009380, 0.009370, 0.009350, 0.009340, 0.009330, 0.009310, 0.009300,
0.009290, 0.009280, 0.009280, 0.009270, 0.009260, 0.009260, 0.009260, 0.009250, 0.009250,
0.009250, 0.009250, 0.009260, 0.009260, 0.009260, 0.009270, 0.008300, 0.008310, 0.008330,
0.008340, 0.008350, 0.008360, 0.008370, 0.008390, 0.008400, 0.008410, 0.008420, 0.008430,
0.008440, 0.008450, 0.008460, 0.008460, 0.008470, 0.008480, 0.008490, 0.008500, 0.008500,
0.008510, 0.008520, 0.008530, 0.008530, 0.008540, 0.008540, 0.008550, 0.008550, 0.008560,
0.008560, 0.008570, 0.008570, 0.008570, 0.008580, 0.008580, 0.008580, 0.008580, 0.008590,
0.008590, 0.008590, 0.008590, 0.008590, 0.008590, 0.008590, 0.008590, 0.008590, 0.008590,
0.008590, 0.008590, 0.008590, 0.008590, 0.008580, 0.008580, 0.008580, 0.008580, 0.008570,
0.008570, 0.008570, 0.008560, 0.008560, 0.008550, 0.008550, 0.008540, 0.010860, 0.010870,
0.010880, 0.010890, 0.010890, 0.010900, 0.010900, 0.010910, 0.010910, 0.010910, 0.010920,
0.010920, 0.010920, 0.010920, 0.010920, 0.010920, 0.010920, 0.010920, 0.010920, 0.010910,
0.010910, 0.010910, 0.010900, 0.010900, 0.010890, 0.010880, 0.010880, 0.010870, 0.010860,
0.010850, 0.010840, 0.010830, 0.010820, 0.010810, 0.010800, 0.010790, 0.010770, 0.010760,
0.010750, 0.010730, 0.010720, 0.010700, 0.010690, 0.010670, 0.010650, 0.010630, 0.010610,
0.010590, 0.010570, 0.010550, 0.010530, 0.010510, 0.010490, 0.010470, 0.010440, 0.010420,
0.010390, 0.010370, 0.010340, 0.010320, 0.010290, 0.010260, 0.010230, 0.010210, 0.011180,
0.011160, 0.011140, 0.011130, 0.011110, 0.011090, 0.011080, 0.011060, 0.011040, 0.011020,
0.011010, 0.010990, 0.010970, 0.010950, 0.010940, 0.010920, 0.010900, 0.010880, 0.010870,
0.010850, 0.010830, 0.010810, 0.010800, 0.010780, 0.010760, 0.010740, 0.010720, 0.010710,
0.010690, 0.010670, 0.010650, 0.010630, 0.010620, 0.010600, 0.010580, 0.010560, 0.010540,
0.010520, 0.010500, 0.010490, 0.010470, 0.010450, 0.010430, 0.010410, 0.010390, 0.010370,
0.010350, 0.010340, 0.010320, 0.010300, 0.010280, 0.010260, 0.010240, 0.010220, 0.010200,
0.010180, 0.010160, 0.010140, 0.010120, 0.010110, 0.010090, 0.010070, 0.010050, 0.010030}

history = begins

nemo_header_test, version 1.0
Thu Oct 14 16:01:04 1999
sensor type = AVIRIS
cal_file = D:\new_data_system\examples\coral_cal_file.asc
image_center_date = { 1996, 3, 23}
image_center_time = { 19, 44, 29.0000}
image_center_long = { 81, 47, 54.0000}
image_center_long_hem = W
image_center_lat = { 24, 36, 44.0000}
image_center_lat_hem = N
image_center_zenith_ang = { 0, 0, 0.000000}
image_center_azimuth_ang = { 0, 0, 0.000000}
sensor_altitude =  19.7680

history = ends
```

## A.2 Output Header Files

### A.2.1 *Mask* Output Header File

Based on the above files, the name of the mask output image will be `./coral_test_mask.img` and the image header file will be `./coral_test_mask.hdr`.

```
ENVI
description = {
  Masks: 0=not masked, 100=masked
  derived from mask;
  Land mask based on NDVI calculation; NDVI calculation uses information
  in history section below (ndvi_wavelengths, ndvi_nbands).
  measurement date: 1996-03-23; measurement time: 19:44:29.000 GMT
  latitude:  24deg 36m 44.000s N ; longitude:  81deg 47m 54.000s W
  view zenith angle:   0deg  0m  0.000s ;
  relative azimuth angle:   0deg  0m  0.000s ;
 }
samples = 614
lines   = 600
bands   = 4
header offset = 0
file type = ENVI Standard
data type = 1
interleave = bsq
byte order = 1
sensor type = AVIRIS
x start = 1
y start = 298
default stretch = 0.000000 100.000000 linear
image_center_date = { 1996, 03, 23}
image_center_time = { 19., 44., 29.000}
image_center_lat = { 24., 36., 44.000}
image_center_lat_hem = N
image_center_long = {  81., 47., 54.000}
image_center_long_hem = W
image_center_zenith_ang = {  0.,  0.,  0.000}
image_center_azimuth_ang = {   0.,  0.,  0.000}
sensor_altitude =    19.768
image_scale_factor = {       1.0000}
image_unscaled_units = { dimensionless}
band names = {
Land Mask, Cirrus Mask, Low Altitude Cloud Mask, Bad Pixel Mask}

history = begins

nemo_header_test, version 1.0
Thu Oct 14 16:01:04 1999
sensor type = AVIRIS
cal_file = D:\new_data_system\examples\coral_cal_file.asc
image_center_date = { 1996, 3, 23}
image_center_time = { 19, 44, 29.0000}
image_center_long = { 81, 47, 54.0000}
image_center_long_hem = W
image_center_lat = { 24, 36, 44.0000}
image_center_lat_hem = N
```

```
image_center_zenith_ang = { 0, 0, 0.000000}
image_center_azimuth_ang = { 0, 0, 0.000000}
sensor_altitude =  19.7680

mask_name = mask for SGI
mask_version = 0.9i 2001-Aug-08-13:00:00 EDT
mask_execution_date = 2001-Aug-08
mask_execution_time = 13:06:56   -0400
mask_input_image_name = /usr/people/mmontes/atrem_cois/coral.bip
mask_which_masks = { land, cirrus, low cloud}
mask_data_directory = ../
mask_output_root_name = ./coral_test
ndvi_scale_factor =  10000.0000
ndvi_land_mask_threshold =      0.0500
refl_land_mask_threshold =     -1.0000
ndvi_wavelengths = { 0.660000, 0.860000 }
ndvi_nbands = {   0,  0 }
cirrus_mask_threshold =      0.0045
cirrus_mask_nbands =   0
cloud_mask_ocean_threshold =      0.1000
cloud_mask_nbands =   0

history = ends
```

### A.2.2 *NDVI* Output Header File

Based on the above files, the name of the mask output image will be `./coral_test_ndvi.img` and the image header file will be `./coral_test_ndvi.hdr`.

```
ENVI
description = {
  NDVI values x 10000.00, refl(0.660
  derived from mask;
  Land mask based on NDVI calculation; NDVI calculation uses information
  in history section below (ndvi_wavelengths, ndvi_nbands).
  measurement date: 1996-03-23; measurement time: 19:44:29.000 GMT
  latitude:  24deg 36m 44.000s N ; longitude:  81deg 47m 54.000s W
  view zenith angle:   0deg  0m  0.000s ;
  relative azimuth angle:   0deg  0m  0.000s ;
 }
samples = 614
lines   = 600
bands   = 3
header offset = 0
file type = ENVI Standard
data type = 2
interleave = bsq
byte order = 1
sensor type = AVIRIS
x start = 1
y start = 298
image_center_date = { 1996, 03, 23}
image_center_time = { 19., 44., 29.000}
image_center_lat = { 24., 36., 44.000}
image_center_lat_hem = N
```

```
image_center_long = {  81., 47., 54.000}
image_center_long_hem = W
image_center_zenith_ang = {  0.,  0.,  0.000}
image_center_azimuth_ang = {   0.,  0.,  0.000}
sensor_altitude =    19.768
image_scale_factor = {   10000.0000}
image_unscaled_units = { dimensionless}
band names = {
NDVI x 10000.00, refl(0.6600) x 10000.00, refl(0.8600) x 10000.00}

history = begins

nemo_header_test, version 1.0
Thu Oct 14 16:01:04 1999
sensor type = AVIRIS
cal_file = D:\new_data_system\examples\coral_cal_file.asc
image_center_date = { 1996, 3, 23}
image_center_time = { 19, 44, 29.0000}
image_center_long = { 81, 47, 54.0000}
image_center_long_hem = W
image_center_lat = { 24, 36, 44.0000}
image_center_lat_hem = N
image_center_zenith_ang = { 0, 0, 0.000000}
image_center_azimuth_ang = { 0, 0, 0.000000}
sensor_altitude =  19.7680

mask_name = mask for SGI
mask_version = 0.9i 2001-Aug-08-13:00:00 EDT
mask_execution_date = 2001-Aug-08
mask_execution_time = 13:06:56   -0400
mask_input_image_name = /usr/people/mmontes/atrem_cois/coral.bip
mask_which_masks = { land, cirrus, low cloud}
mask_data_directory = ../
mask_output_root_name = ./coral_test
ndvi_scale_factor =  10000.0000
ndvi_land_mask_threshold =      0.0500
refl_land_mask_threshold =     -1.0000
ndvi_wavelengths = { 0.660000, 0.860000 }
ndvi_nbands = {   0,  0 }
cirrus_mask_threshold =      0.0045
cirrus_mask_nbands =   0
cloud_mask_ocean_threshold =      0.1000
cloud_mask_nbands =   0

history = ends
```